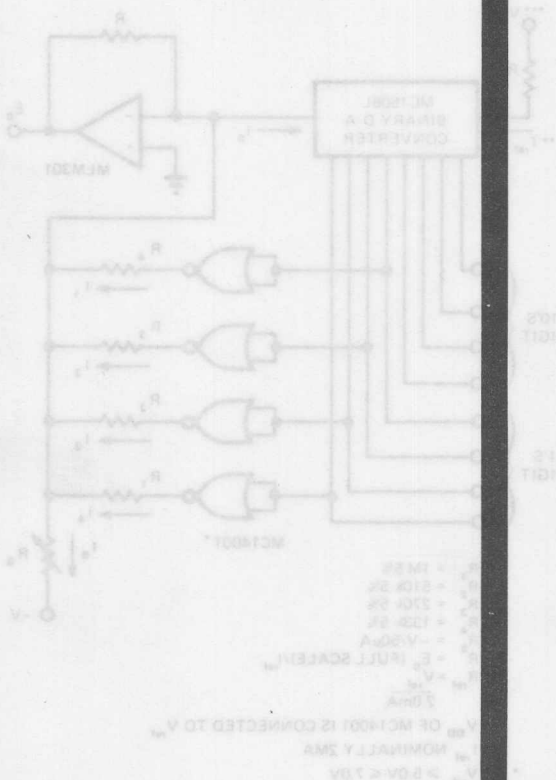


BINARY D/A CONVERTERS CAN PROVIDE BCD-CODED CONVERSION

Prepared by:
Tim Henry
 Motorola Semiconductor

Reprinted with permission of EDN Magazine,
 August 5, 1973

This note describes the application and use of integrated circuit D/A converters for use in providing a BCD-coded conversion. The technique is illustrated using a 2-1/2 digit digital voltmeter.



MOTOROLA Semiconductor Products Inc.

Binary D/A converters can provide BCD-coded conversion

You can use IC D/A converters, even though they're binary coded, to do BCD-to-analog conversion. It just takes a few extra parts.

Monolithic digital-to-analog (D/A) converters have become very popular because of their versatility and low cost. They have one limitation, however, and that is that they are all binary coded. Thus, they cannot be used directly for the many applications where a binary-coded decimal (BCD) conversion is required. It is possible, though, by adding some external components to make a binary-coded D/A converter perform as a BCD code converter.

Two-digit converter

A 2-digit, BCD-coded voltage-output D/A converter is shown in Fig. 1. To understand its operation, note the 4-bit binary code and corresponding one digit of BCD code shown in Table 1. The BCD code and the 0-to-9 of the binary code are exactly the same. The 4-bit binary code sequences through all 16 steps before the next most-significant bit (fifth bit) increments one step. However, when counting in BCD the 4-bit code will only sequence through ten steps before the next most-significant bit increments once. This means that 10 least-significant bit steps in BCD equal 16 least-significant bit steps in binary, assuming the next most-significant bit of both codes is the same magnitude.* Therefore, by making the four least-significant bits of the binary-weighted D/A converter appear larger, such that 10 least-significant bit steps equal the magnitude of the next significant bit, a BCD-coded D/A converter can be effectively produced.

To make the four least-significant bits of the D/A appear larger than normal (Fig. 1), current from the node connecting the output of the D/A converter and the virtual ground of the op amp is used. This virtual ground of the op amp provides a very good summing junction.

In implementing the technique it was found that the hardware available made it much easier to switch currents into the node than out of it. This problem was circumvented by taking a constant current out of the node and switching currents into it.

The output of the binary D/A is a current sink,

with the amount of current depending on the reference current input, I_{ref} , and the digital word on its input lines. This is empirically given by:

$$I_o = I_{ref} \left(\frac{X}{256} \right)$$

$$I_{ref} = \frac{V_{ref}}{R_{ref}}$$

where I_o is the output current, I_{ref} is the reference current and x is the digital word input.

The op-amp feedback resistor changes analog current I_o to an analog voltage, e_o , where:

$$e_o = I_o R$$

Thus, e_o is directly proportional to I_o , namely R times.

The outputs of the CMOS NOR gates appear as voltage sources, with 750Ω output impedances. In

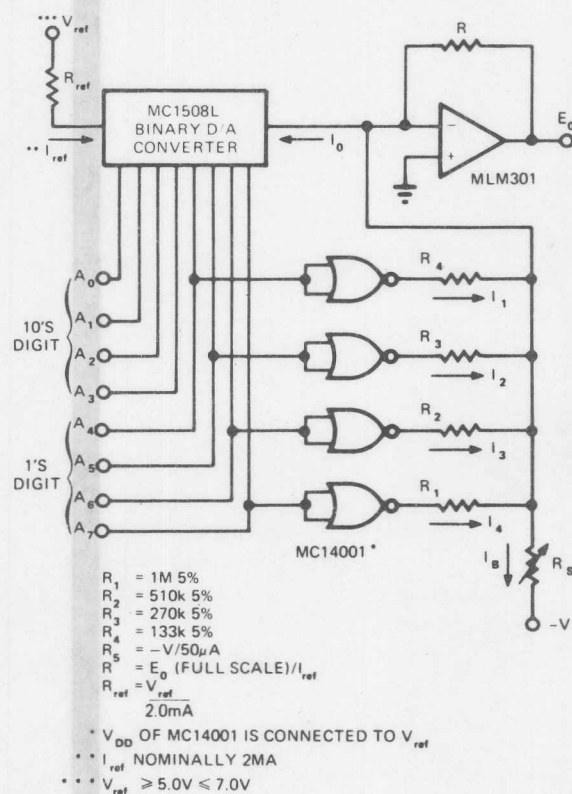


Fig. 1—Two-digit BCD D/A converter uses a binary-coded monolithic D/A converter as its basic conversion element.

Circuit diagrams external to Motorola products are included as a means of illustrating typical semiconductor applications; consequently, complete information sufficient for construction purposes is not necessarily given. The information in this Application Note has been carefully checked and is believed to be entirely reliable. However, no responsibility is assumed for inaccuracies. Furthermore, such information does not convey to the purchaser of the semiconductor devices described any license under the patent rights of Motorola Inc. or others.

TABLE 1 — 4-BIT BINARY VS 1-DIGIT BCD

	BINARY	BCD
0	0 0 0 0	0 0 0 0
1	0 0 0 1	0 0 0 1
2	0 0 1 0	0 0 1 0
3	0 0 1 1	0 0 1 1
4	0 1 0 0	0 1 0 0
5	0 1 0 1	0 1 0 1
6	0 1 1 0	0 1 1 0
7	0 1 1 1	0 1 1 1
8	1 0 0 0	1 0 0 0
9	1 0 0 1	1 0 0 1
10	1 0 1 0	_____
11	1 0 1 1	_____
12	1 1 0 0	_____
13	1 1 0 1	_____
14	1 1 1 0	_____
15	1 1 1 1	_____

other words, when the output of a gate is LOW, it looks like a 750Ω resistor to ground. When the output is HIGH, a gate looks like a 750Ω resistor to V_{ref} .

If the output of a gate is LOW, the voltage across its resistor from the output to the virtual ground (R_1 through R_4) is approximately zero. However, when the output of a gate is HIGH, the voltage across its resistor is V_{ref} . Therefore, the current into the summing node is given by:

$$I_1 = \frac{V_{ref}}{R_1}$$

$$I_2 = \frac{V_{ref}}{R_2}$$

$$I_3 = \frac{V_{ref}}{R_3}$$

$$I_4 = \frac{V_{ref}}{R_4}$$

Since the NOR gates function as inverters for the digital word input, the outputs of the inverters are going to be normally HIGH for a ZERO on the input line. Now, if the currents are set so that

$$I_B = I_1 + I_2 + I_3 + I_4$$

when the input word is all ZEROs, no net current will be taken from or added to the summing node. If one of the four least-significant bits is turned ON, the output of that inverter goes LOW

and the current through the resistor to the summing junction is zero. This requires a net current, equal to the amount that was being injected into the summing junction by the resistor, to be drawn out of the summing junction, causing that bit of the D/A to appear larger than it really is. The other three bits work in exactly the same manner.

The only problem now is to determine the values of the resistors for proper operation.

Since the output of the D/A is a direct function of the reference current, it follows that the added currents must be also. The most-significant current of the D/A is $I_{ref}/2$. The second most-significant bit current is $I_{ref}/4$ and so on. **Table 2** shows the value for the current of each bit of the D/A and also gives the values needed for BCD operation. The difference between the binary value and the BCD value must be taken from the node to make the binary-weighted D/A behave as though it were BCD coded.

It is interesting to note that making the D/A converter act as if it were BCD coded could also be achieved if the four most-significant bits of the D/A were made to look smaller than normal. The reason the system is set up as described is that the least-significant bit currents do not have to be as accurate as the most-significant bit currents. This means that the magnitude of the current subtracted from the summing junction using the least-significant bits is not critical.

Determining allowable current error

If the BCD-coded D/A converter is to be

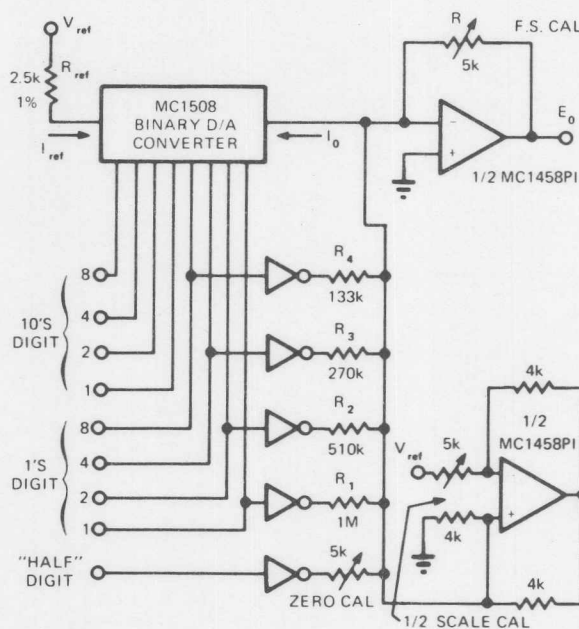


Fig. 2—Addition of a current source and current switch expands the 2-digit BCD converter of Fig. 1 into a 2-1/2-digit converter.



and the current through the resistor to the summing junction is zero. This requires a net current equal to the amount that was being injected into the summing junction by the resistor, to be drawn out of the summing junction, causing that bit of the D/A to appear larger than it really is. The other three bits work in exactly the same manner.

The only problem now is to determine the values of the resistors for proper operation.

Since the output of the D/A is a direct function of the reference current, it follows that the added current must be also. The most significant current of the D/A is $I_{1/4}$. The second most-significant bit current is $I_{1/8}$ and so on. Table 2 shows the value for the current of each bit of the D/A and also gives the values needed for BCD operation. The difference between the binary value and the BCD value must be taken from the node to make the binary-weighted D/A behave as though it were BCD-coded.

It is interesting to note that making the D/A converter act as if it were BCD-coded could also be achieved if the four most-significant bits of the D/A were made to look smaller than normal. The reason the system is set up as described is that the least-significant bit currents do not have to be as accurate as the most-significant bit currents. This means that the magnitude of the current subtracted from the summing junction using the least-significant bits is not critical.

Determining allowable current error

If the BCD-coded D/A converter is to be

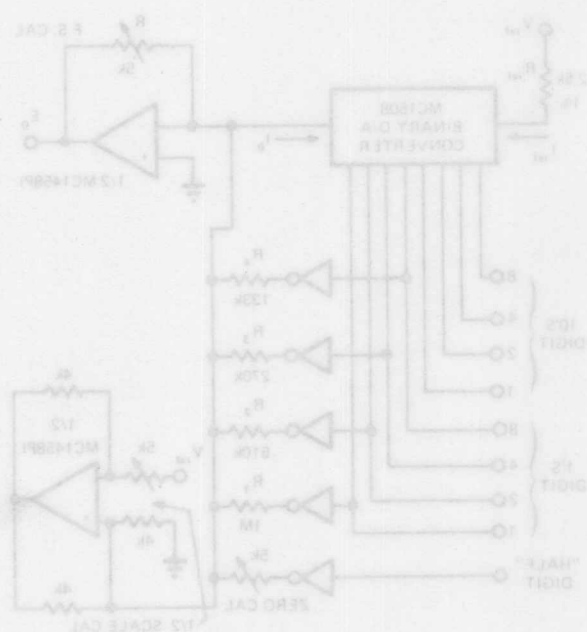


Fig. 2—Addition of a current source and current switch expands the 3-bit BCD converter of Fig. 1 into a 3-1/2-digit converter.

TABLE 1—4-BIT BINARY VS 1-DIGIT BCD

BINARY	BCD
0 0 0 0	0 0 0 0
0 0 0 1	0 0 0 1
0 0 1 0	0 0 1 0
0 0 1 1	0 0 1 1
0 1 0 0	0 1 0 0
0 1 0 1	0 1 0 1
0 1 1 0	0 1 1 0
0 1 1 1	0 1 1 1
1 0 0 0	1 0 0 0
1 0 0 1	1 0 0 1
1 0 1 0	1 0 1 0
1 0 1 1	1 0 1 1
1 1 0 0	1 1 0 0
1 1 0 1	1 1 0 1
1 1 1 0	1 1 1 0
1 1 1 1	1 1 1 1

other words, when the output of a gate is LOW, it looks like a 750Ω resistor to ground. When the output is HIGH, a gate looks like a 750Ω resistor to V_{cc} .

If the output of a gate is LOW, the voltage across its resistor from the output to the virtual ground (R_i through R_j) is approximately zero. However, when the output of a gate is HIGH, the voltage across its resistor is V_{cc} . Therefore, the current into the summing node is given by:

$$I = \frac{V_{cc}}{R_1} + \frac{V_{cc}}{R_2} + \frac{V_{cc}}{R_3} + \frac{V_{cc}}{R_4}$$

Since the NOR gates function as inverters for the digital word input, the outputs of the inverters are going to be normally HIGH for a ZERO on the input line. Now, if the currents are set so that

$$I = I_1 + I_2 + I_3 + I_4$$

when the input word is all ZEROs, no net current will be taken from or added to the summing node. If one of the four least-significant bits is turned ON, the output of that inverter goes LOW



MOTOROLA Semiconductor Products Inc.

Printed in Switzerland

accurate, the maximum amount that any of the bit currents can deviate from the ideal value is 50% of the least-significant bit current. Since the least-significant bit current is larger for BCD than for binary, a less accurate D/A is required to give 2-digit BCD accuracy (100 steps) than for 8-bit binary (256 steps).

Table 2 shows that the value of the least-significant bit current for BCD is 0.125 mA. So to give the required accuracy, each bit current must not deviate from the ideal value by more than ± 0.00625 mA, or 6.25 μ A.

For an 8-bit binary D/A, assuming a 2 mA ladder current, the maximum error that any bit may have is 3.9 μ A. Assuming the 8-bit D/A to have the worst allowable error on each of the least-significant bits, that leaves an error of 6.25 μ A minus 3.9 μ A, or 2.35 μ A to be introduced by the injected currents. In other words, to insure that the 2-digit BCD-coded D/A is accurate, the injected currents must be kept within ± 2.35 μ A of their ideal value.

It is easy now to determine the accuracy required for the injected currents. The percentage of accuracy is simply the amount of deviation allowed, 2.35 μ A, divided by the amount of injected current. This is given by:

$$\% \text{ allowable error} = \frac{2.35 \mu\text{A}}{\text{amount current injected}} \times 100$$

Using this formula, the injected currents' allowable errors are:

LSB;	50%
2nd LSB;	25%
3rd LSB;	12.5%
4th LSB;	6.25%

This shows that 5% tolerance resistors are more than adequate. **Fig. 1** gives the resistor values for a 5.0V reference voltage. These values are sufficient to neglect the output impedance of the NOR gates.

Calibration of the circuit of **Fig. 1** is as follows: First, V_{ref} or R_{ref} is adjusted to give a half-scale

TABLE 2 - BIT CURRENTS

BIT	NUMBER	BINARY	BCD	DIFFERENCE
A ₀	MSB	1.000MA*	1.000MA*	0
A ₁	2ND	0.5000	0.5000	0
A ₂	3RD	0.2500	0.2500	0
A ₃	4TH	0.1250	0.1250	0
A ₄	5TH	0.0625	0.1000	0.0375
A ₅	6TH	0.0312	0.0500	0.0188
A ₆	7TH	0.0158	0.0250	0.0094
A ₇	LSB	0.0078	0.0125	0.0047

* I_{ref} = 2.0000mA

reading of e_0 with only the most-significant bit ON. Next, with all bits turned OFF, R_5 is adjusted so that e_0 is zero. The D/A is now calibrated.

2-1/2-digit converter

In many applications, a 2-1/2-digit BCD-coded converter is desired. That is, a circuit that will count to 199 rather than to 99. Once the basic 2-digit circuit has been designed, it is relatively easy to add the half digit. **Fig. 2** shows such a 2-1/2-digit circuit. It is identical to the 2-digit configuration, except for the addition of a current source and a current switch to produce the 1/2 digit.

In operation, the circuit should sequence through steps 0 to 99 while the 1/2 digit is LOW and through steps 100 to 199 while the 1/2 digit is HIGH. This means that the 1/2-digit current is equal to 100 least-significant bits of current. If the least-significant bit current in the 2-1/2-digit circuit is the same as the least-significant bit current in the 2-digit circuit previously described, the value for the 1/2 digit is 1.250 mA (100 \times 0.0125 mA). Therefore, the circuit will act as a 2-1/2-digit BCD-coded D/A if the 1/2-digit switch sinks zero current for the first 100 counts (0 to 99) and sinks 1.250 mA for the second 100 counts (100 to 199).

The 1/2-digit current is added in the same manner as the least-significant bit currents were added in the 2-digit system. Namely, a constant current, I_B , is sunk from the summing node, and the various currents, I_1 through I_5 , are switched into the node. For the 2-1/2-digit circuit, I_B is obtained by:

$$I_B = I_1 + I_2 + I_3 + I_4 + I_5$$

where I_1 through I_4 are identical with their 2-digit system counterparts.

The tolerance of I_B to assure that the D/A remains accurate is 2.35 μ A. This is the same value as that derived in the 2-digit section. Since I_B is much larger for the 2-1/2-digit system, the percentage tolerance of I_B is much more critical. If the values for I_1 through I_4 shown in **Table 2** are summed with the value of I_5 , we have;

$$I_B = 4.7 \mu\text{A} + 9.4 \mu\text{A} + 18.8 \mu\text{A} + 37.5 \mu\text{A} + 1250 \mu\text{A} = 1320 \mu\text{A}.$$

If I_B can only vary ± 2.35 μ A, its tolerance is $\pm 0.18\%$.

In the 2-digit system, current I_B could be produced simply by a resistor from the summing node to the negative supply because the summing node is a virtual ground. Once the system is calibrated by adjusting R_5 , current I_B is constant except for changes caused by variations in the negative supply voltage.

In the 2-1/2-digit system, a circuit is needed that will sink current from the summing node without being referenced to the negative supply voltage.

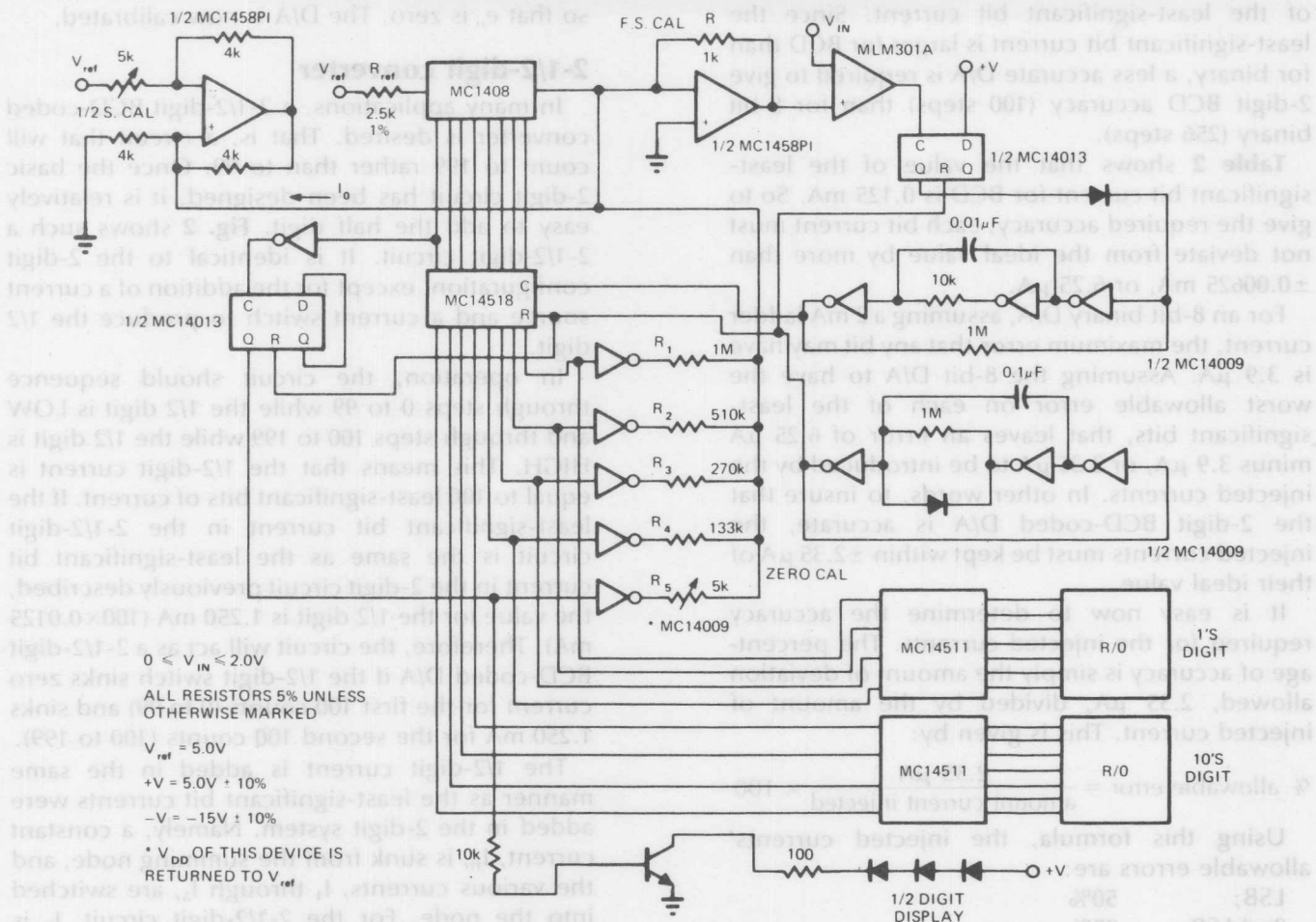


Fig. 3—2-1/2-digit DVM uses a binary D/A converter to accomplish conversion of a BCD digital input signal.

One circuit that fills the need is the operational-amplifier current source shown in Fig. 2.

There is one other difference between the basic 2-digit system and the 2-1/2-digit circuit. In the 2-1/2-digit implementation, a CMOS hex-inverter is used instead of a quad 2-input NOR gate package. This is because the 2-1/2-digit system requires five switches.

The 2-1/2-digit system is calibrated as follows:

- Attach an accurate DVM to the output, e_o . With all inputs LOW, adjust ZERO CAL potentiometer for a zero reading of e_o .
- Put a HIGH input only on the most-significant bit of the monolithic D/A (0 1000 0000), then read and record e_o .
- Put a HIGH input only on the 1/2 digit and adjust the 1/2 SCALE CAL potentiometer to

give exactly 1.25 times the reading of the previous step.

- With all inputs LOW, readjust the ZERO CAL potentiometer for an e_o of exactly 0V.
- Finally, input the BCD word for 199, (1 1001 1001) and adjust the F.S. CAL for the desired full-scale reading.

DVM shows application

Fig. 3 shows how the technique can be incorporated into a 2-1/2-digit digital voltmeter. The circuit uses the staircase type of conversion, with the staircase being produced by the 2-1/2-digit BCD-coded D/A and the BCD counters. The MLM301A is used as a comparator to compare the staircase to the input signal. □